

计算机组成原理

计算机组成原理课程组

(刘旭东、高小鹏、肖利民、牛建伟、梁钟治)

第四讲：主存储器

- 一. 存储系统概述
- 二. 存储单元电路
- 三. 存储器芯片结构
- 四. 存储器扩展

存储系统概述

❖ 存储器分类

➢ 按介质分类：

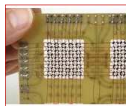
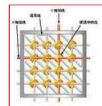
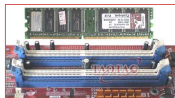
- 半导体存储器（易失性）
- 磁介质存储器（非易失性）
- 光盘存储器（非易失性）

➢ 按访问方式分类：

- 随机访问存储器（Random Access Memory—RAM）
- 只读存储器（Read Only Memory—ROM）
- 顺序访问存储器（Tape）
- 直接访问存储器（Disk）

➢ 按功能分类：

- 高速缓冲存储器
- 主存储器
- 辅助存储器
- 控制存储器



存储系统概述

❖ 存储器的性能指标

➢ 访问时间（Access Time）： T_A

- 随机访问存储器：访问时间指读或写操作所用时间，即从给定地址到存储器完成读或写操作所需时间。
- 其他类型：指将读写机构定位到目标位置所需的时间。

➢ 存储周期（Cycle Time）： T_C

- 仅对RAM而言，指两次访问存储单元间的最小时间间隔。
- $T_C > T_A$

➢ 带宽（Bandwidth）/ 数据传输率（Transfer Rate）

- 一般的随机访问存储器： $1 / \text{Cycle Time}$;

- 其他类型： $T_N = T_A + N/R$

T_N : 读写N Bits所需的平均时间

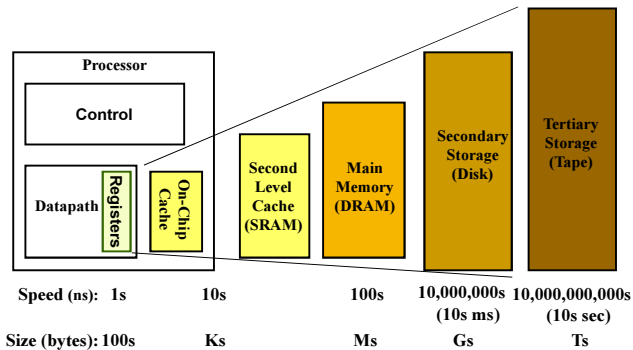
T_A : 访问时间

N: N Bits

R: 存储部件的数据传输率 (bits / s)

存储系统概述

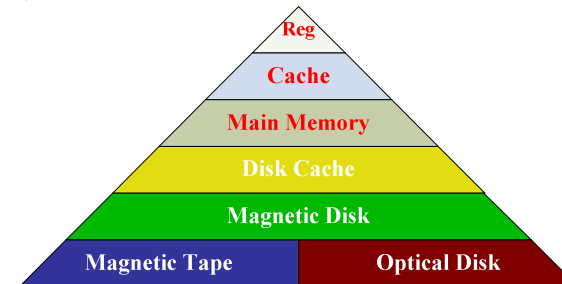
❖ 存储器的层次/性能特征



速度、容量、成本

存储系统概述

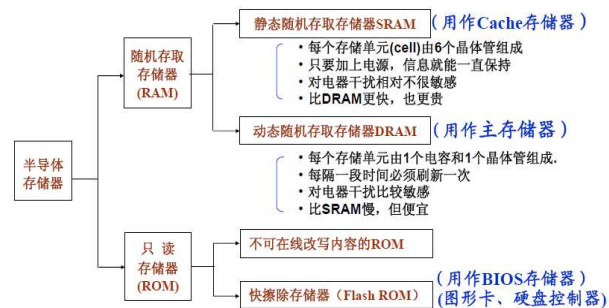
❖ 存储器的层次结构



二级存储系统指：高速缓冲存储器（Cache）+主存储器

三级存储系统指：Cache+主存+辅存（虚拟存储器）

半导体存储器



第四讲：主存储器

- 一. 存储系统概述
- 二. 存储单元电路
- 三. 存储器芯片结构
- 四. 存储器扩展

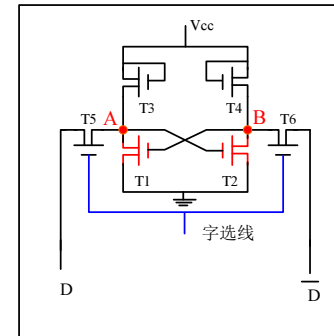
2.1 存储单元电路

❖ 基本要求

- 具有两种稳定（或半稳定）状态，用来表示二进制的 1 和 0；
- 可以实现状态写入（或设置）；
- 可以实现状态读取（或感知）。

2.1 SRAM存储单元电路

❖ SRAM存储单元电路（六管单元电路）



MOS管功能:

T1, T2: 工作管;
T3, T4: 负载管;
T5, T6: 门控管;

稳定状态:

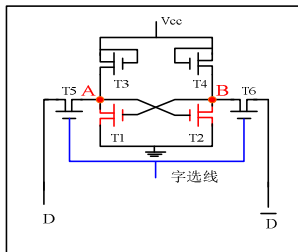
“1”: T1 截止, T2 导通
“0”: T2 截止, T1 导通

保持状态:

字选线低电平, T5 和 T6 截止, 内部保持稳定。

2.1 SRAM存储单元电路

❖ SRAM存储单元电路工作原理（读出）



稳定状态:

“1”: T1 截止, T2 导通
“0”: T2 截止, T1 导通

保持状态:

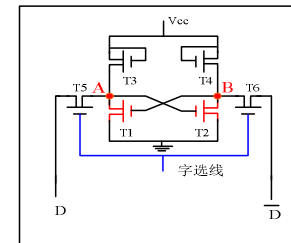
字选线低电平, T5 和 T6 截止, 内部保持稳定。

读出操作:

- 输入条件: 字选线高电平
- T5和T6导通, 如果存储单元原来保存信息是“1”, D线则“读出”了内部状态 (A点电平) 则为高, 否则为低。

2.1 SRAM存储单元电路

❖ SRAM存储单元电路工作原理（写入）



稳定状态:

“1”: T1 截止, T2 导通
“0”: T2 截止, T1 导通

保持状态:

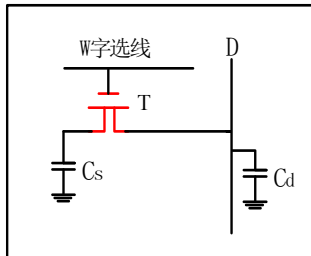
字选线低电平, T5 和 T6 截止, 内部保持稳定。

写入操作:

- 写 1: D线高电平, \bar{D} 线低电平, 字选线高电平, T5 和 T6 导通, T1截止, T2导通, 写入 1。
- 写 0: D线低电平, \bar{D} 线高电平, 字选线高电平, T5 和 T6 导通, T2截止, T1导通, 写入 0。

2.2 DRAM存储单元电路

❖ DRAM存储单元电路（单管单元电路）

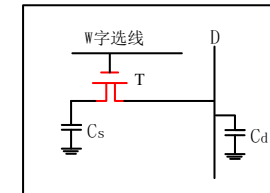


- C_s 电容 \ll C_d 电容
- C_s 上有电荷表示 ‘1’
- C_s 上无电荷表示 ‘0’
- 保持状态：字选线低电平，T截止，理论上内部保持稳定状态。

注意：在保存二进制信息“1”的状态下， C_s 有电荷，但 C_s 存在漏电流， C_s 上的电荷会逐渐消减，状态不能长久保持，在电荷泄漏到威胁所保存的数据性质之前，需要补充所泄漏的电荷，以保持数据性质不变。这种电荷的补充称之为刷新（或再生）。

2.2 DRAM存储单元电路

❖ DRAM存储单元电路工作原理（读出）

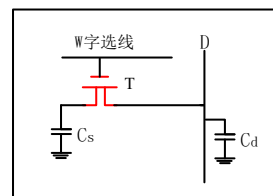


读出时：D线先预充电到 $V_{pre}=2.5V$ ，然后字选线高电平，T导通

- 若电路保存信息1， $V_{cs}=3.5V$ ，电流方向从单元电路内部向外；
- 若电路保存信息0， $V_{cs}=0.0V$ ，电流方向从外向单元电路内部；
- 因此根据数据线上电流的方向可判断单元电路保存的是1还是0。
- 读出过程实际上是 C_s 与 C_d 上的电荷重新分配的过程，也是 C_s 与 C_d 上的电压重新调整的过程。 C_d 上的电压，即是D线上的电压。

2.2 DRAM存储单元电路

❖ DRAM存储单元电路工作原理（写入）

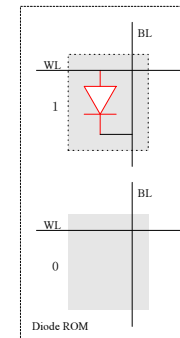


写入操作：D线加高电平（1）或低电平（0），字选择线置高电平，T导通；

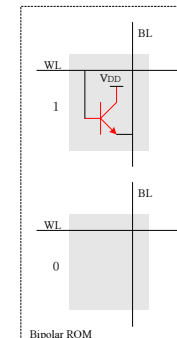
- 写1时，D线高电平，对 C_s 充电；
- 写0时，D线低电平， C_s 放电；

2.3 ROM存储单元电路

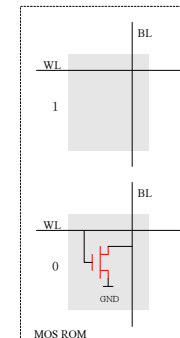
❖ 固定掩膜ROM单元电路



含二极管的电路表示1，不含电路表示0



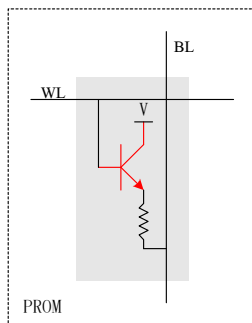
含三极管的电路表示1，不含电路表示0



含MOS管的电路表示1，不含电路表示0

2.3 ROM存储单元电路

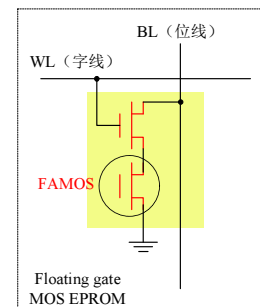
❖ 可编程的PROM单元电路



- 出厂时所有位均为1。
- 编程时（写入数据），对写0的单元加入特定的大电流，熔丝被烧断，变为另一种表示0的状态，且不可恢复。
- 工作时，加入正常电路。

2.3 ROM存储单元电路

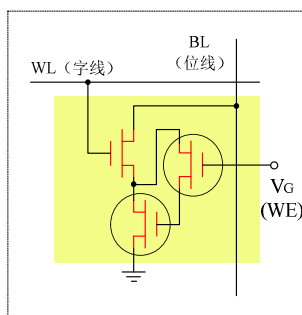
❖ 紫外线擦除可编程的EPROM单元电路



- 出厂时所有位均为1，FAMOS（浮空栅极MOS）G极无电荷，处于截止状态。
- 编程时（写入数据），对写0的单元加入特定的电压，FAMOS上的G极与D极被瞬时击穿，大量电子聚集到G极上，撤销编程电压后，G极上的聚集的电子不能越过隔离层，FAMOS导通，表示0。
- 工作时，加入正常电压，FAMOS的状态维持不变。
- 擦除时，用紫外线照射，FAMOS聚集在G极上的电子获得能量，越过隔离层泄漏，FAMOS恢复截止状态。

2.3 ROM存储单元电路

❖ EEPROM单元电路

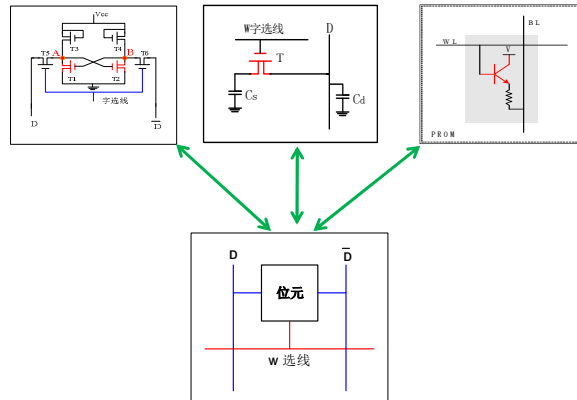


- 与EPROM相似，它是在EPROM基本单元电路的浮空栅的上面再生成一个浮空栅，前者称为第一级浮空栅，后者称为第二级浮空栅。第二级浮空栅引出一个电极，接某一电压 V_G 。
- 若 V_G 为正电压，第一浮空栅极与漏极之间产生隧道效应，使电子注入第一浮空栅极，即编程写入。
- 若使 V_G 为负电压，强使第一级浮空栅极的电子散失，即擦除。擦除后可重新写入。

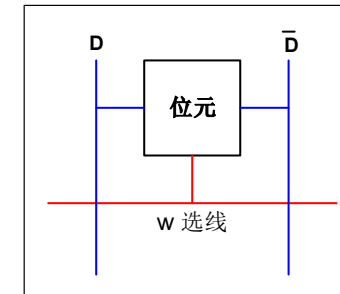
❖ 主存储器——存储单元电路

- ❖ 存储单元电路的概念及其基本条件（0/1状态、读出、写入）
- ❖ SRAM存储单元电路工作原理（六管、触发器）
- ❖ DRAM存储单元电路工作原理（单管、电容充放电、需要刷新）
- ❖ ROM存储单元电路的工作原理（ROM、PROM、EPROM、EEPROM）

❖ 存储单元的符号表示



❖ 存储单元电路: 可存储1位(1bit)二进制代码



存储单元电路 (存储位元) → 存储芯片 → 存储器

第四讲：主存储器

- 一. 存储系统概述
- 二. 存储单元电路
- 三. 存储器芯片结构
- 四. 存储器扩展

存储芯片内部结构

❖ 存储芯片容量的基本描述 (字单元数 × 每个字单元的位数)

➤ $1K \times 2$: 1024 个字单元, 每个字单元2 位 (二进制位)

意味着任一时刻可以 (也只能) 访问1024个独立字单元中的任意一个, 每次读写的数据位数是一个字单元的容量 (2位)

对于 $1K \times 2$ 的存储芯片:

有多少个存储位元? 共1K个 (1024个) 字单元, 每个字单元2位 2048

需多少条地址线? 按字单元寻址, 1024个 (2^{10} 个) 字单元 10

需要多少条数据线? 一次访问一个字单元, 每个字单元是2位 2

❖ $64K \times 8$: 65536 (64K) 个字单元, 每个字单元8位

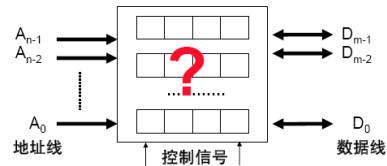
有多少个存储位元? 需要多少条地址线? 多少条数据线?

存储芯片容量的描述: $2^n \times m$ (字单元数 \times 每个字单元的位数)

❖ 存储位元: $2^n \times m$ 个

❖ 地址线: n 位 $\rightarrow 2^n$ 个字单元, $A_{n-1}..A_0$ 单向

❖ 数据线: m 位 $\rightarrow m$ 位/字单元, $D_{m-1}..D_0$ 双向

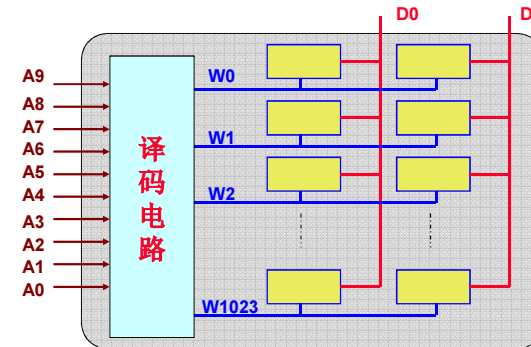


一维地址结构(线选法) / 二维地址结构 (重合法)

3.1 存储芯片内部结构

❖ 存储芯片结构 (一维地址结构)

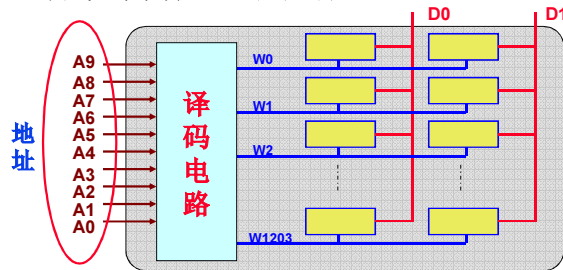
1024×2 : 1024 个字单元, 每个字单元 2 个二进制位。



3.1 存储芯片内部结构

❖ 问题: 如何识别这些字单元?

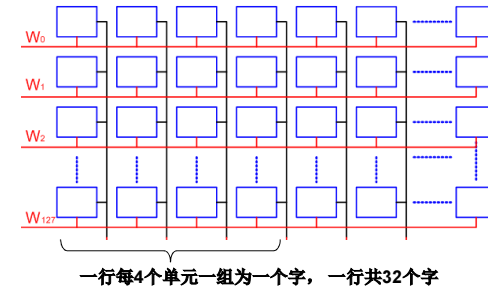
- 1024×2 : 1024 个字单元, 需要1024个不同的标识。
- 地址编码: 译码电路使得字选择线 W_i 处于工作状态的输入信号 (二进制信号), 称为 W_i 所选中字单元的地址编码 (简称地址)。
- 对于每一个字单元, 地址是唯一的。



3.1 存储芯片内部结构

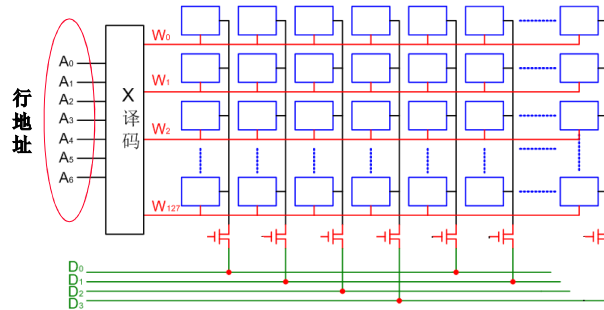
❖ 二维地址结构 (SRAM)

- 芯片示例: 4096×4 (4096 个字, 每个字 4 位)
- $4096 \times 4 = 2^{14}$ 个位单元
- 存储矩阵: $2^7 \times 2^7$ (128行 \times 128列)



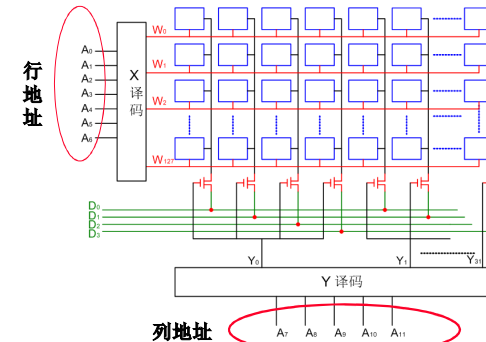
3.1 存储芯片内部结构

- ❖ 二维地址结构 (SRAM) : 4096×4 : 4096 个字, 每个字 4 位
 - 存储矩阵: $2^7 \times 2^7$ (128 行 \times 128 列)
 - 行译码: 行地址 7 位, 一行含 32 个字共 128 位, 任一时刻只有 1 个字 (4 位数据线) 被选中。



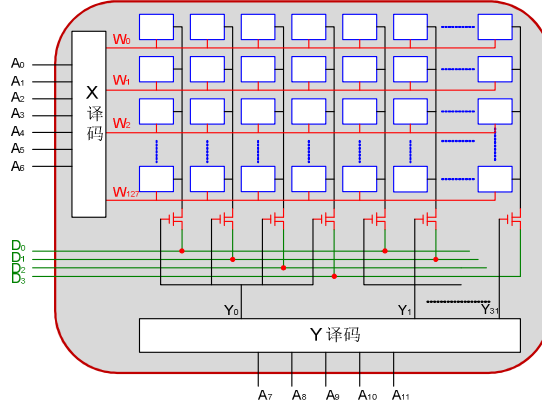
3.1 存储芯片内部结构

- ❖ 二维地址结构 (SRAM) : 4096×4 : 4096 个字, 每个字 4 位
 - 存储矩阵: $2^7 \times 2^7$ (128 行 \times 128 列)
 - 一行包括 32 个字, 要进行 32 选 1 的译码 (Y 译码), 列地址 5 位



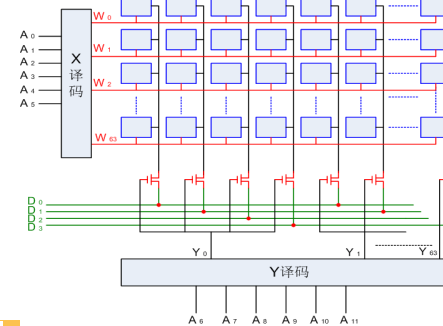
3.1 存储芯片内部结构

- ❖ 二维地址结构 (SRAM) : 4096×4 : 4096 个字, 每个字 4 位。



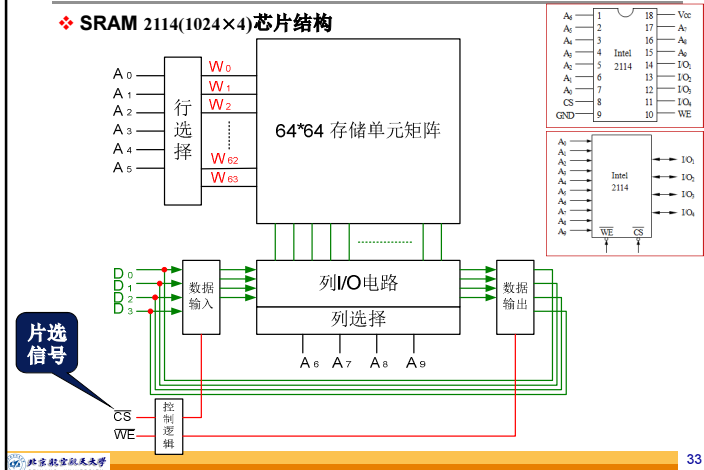
3.1 存储芯片内部结构

- ❖ DRAM 芯片结构
 - 芯片示例: 4096×4 DRAM (4096 个字, 每个字 4 位)
 - 4096 个字 = 2^{12} , 12 位地址
 - DRAM 芯片封装的特殊: 行列地址管脚复用, 行列地址各 6 位。
 - 存储矩阵: $2^6 \times (2^6 \times 4)$ (64 行 \times 256 列)



存储芯片结构示例

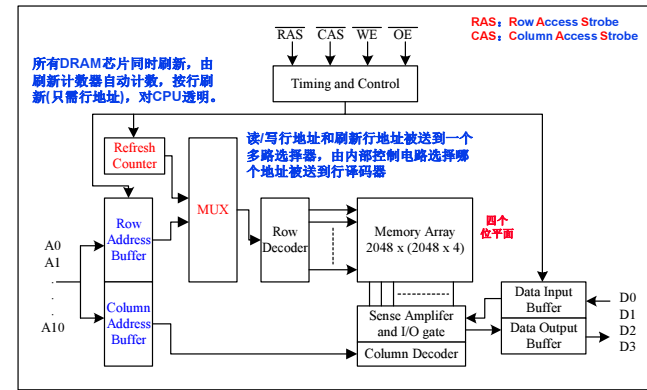
❖ SRAM 2114(1024×4) 芯片结构



33

存储芯片结构示例

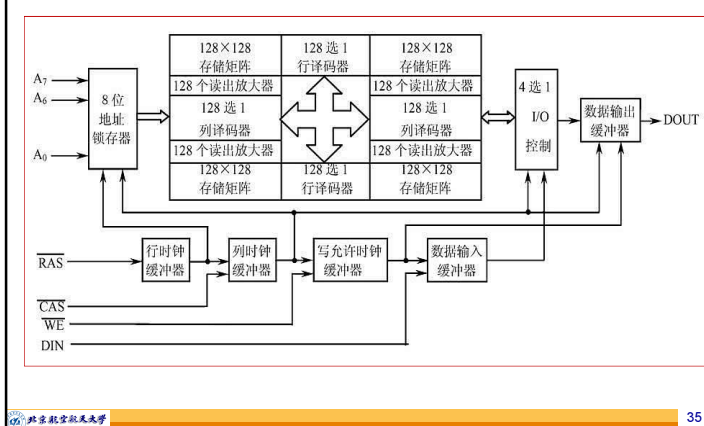
❖ DRAM 4M×4 DRAM芯片结构(内部包含刷新电路)



34

存储芯片结构示例

❖ DRAM 2164A 芯片结构 (64K×1)



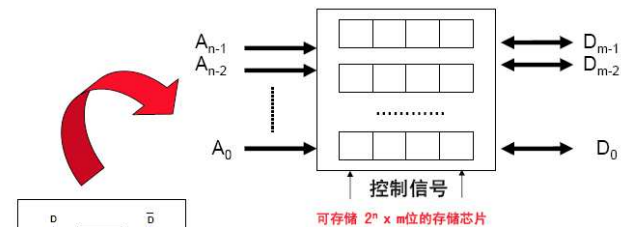
35

第四讲：主存储器

- 一. 存储系统概述
- 二. 存储单元电路
- 三. 存储器芯片结构
- 四. 存储器扩展

36

存储单元电路→ 存储芯片



可存储1位(1bit)的存储单元电路

存储单元→存储电路芯片→存储器?

存储器芯片的扩展

❖ 单片存储器芯片不能满足存储系统的需求

❖ 存储扩展

➢ 位扩展: ($2^n \times m$)

存储器芯片提供的字空间满足整个存储空间的字空间要求, 2^n 够
但存储器芯片的位空间不能满足要求, m 不够

方法: 多个存储器芯片的数据位空间拼在一起

➢ 字扩展: ($2^n \times m$)

存储器芯片提供的字空间不能满足整个存储空间的字空间要求, 2^n 不够
但存储器芯片的位空间满足要求, m 够

方法: 多个存储器芯片的字空间拼在一起

➢ 混合扩展: ($2^n \times m$)

存储器芯片提供的字空间不能满足整个存储空间的字空间要求, 2^n 不够
位空间也不能满足要求, m 不够

方法: 综合运用字扩展和位扩展

4.1 存储器芯片的扩展 (位扩展)

例: 1Kx4 SRAM芯片构成1Kx8的存储器

➢ 1Kx4 芯片管脚:

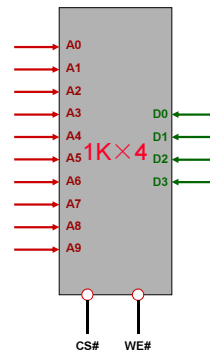
- 10个地址管脚 $A_9 \sim A_0$
- 4个数据管脚 $D_3 \sim D_0$
- 1个片选输入管脚 $CS\#$
- 1个读写控制管脚 $WE\#$
- 芯片地址空间: $000H \sim 3FFH$

➢ CPU访问存储器需提供:

- 地址总线10根: $AB_9 \sim AB_0$
- 数据总线8根: $DB_7 \sim DB_0$
- 读写控制信号: $MemW$
- 存储器地址空间: $000H \sim 3FFH$

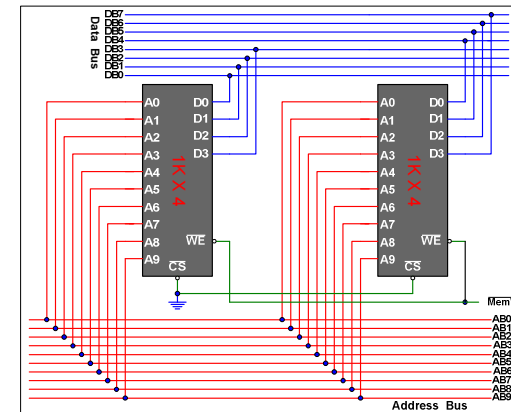
➢ 需要芯片: $(1K \times 8) / (1K \times 4) = 2$ 片

- 地址管脚: 都连接到 $AB_9 \sim AB_0$
- 数据管脚: 分别连接到 $DB_7 \sim DB_4$ 和 $DB_3 \sim DB_0$
- 芯片读写控制管脚: 连接 $MemW$



4.1 存储器芯片的扩展 (位扩展)

例: 1Kx4的SRAM存储芯片构造1Kx8的存储器



4.2 存储器芯片的扩展（字扩展）

例：1Kx8 SRAM芯片构成4Kx8的存储器

➤ 1Kx8 芯片管脚：

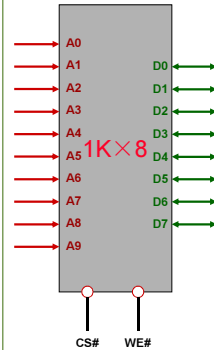
- 10个地址管脚 A9~A0
- 8个数据管脚 D7~D0
- 1个片选输入管脚 CS#
- 1个读写控制管脚 WE#
- 芯片地址空间：000H~3FF H

➤ CPU访问存储器需提供：

- 地址总线12根：AB11~AB0
- 数据总线8根：DB7~DB0
- 读写控制信号：MemW
- 存储器地址空间：000H~FFF H

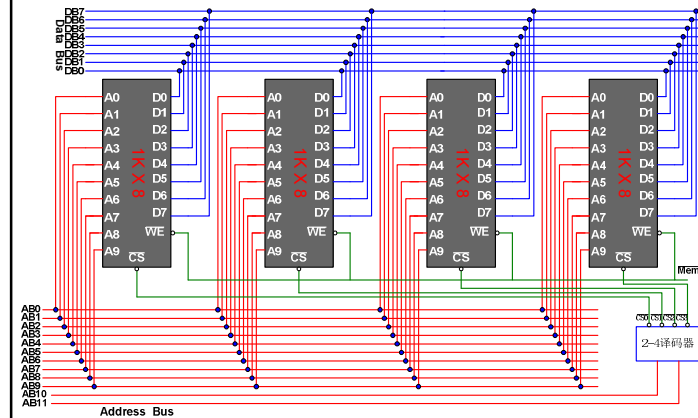
➤ 需要芯片数：(4Kx8)/(1Kx8) = 4片

- 地址管脚：都连接到AB9~AB0
- 数据管脚：都连接到DB7~DB0
- 芯片读写控制管脚：连接MemW
- 一个2-4译码器产生4个片选信号
- 译码器输入：AB11~AB10
- 译码器输出：分别接4个芯片片选管脚



4.2 存储器芯片的扩展（字扩展）

例：1Kx8 SRAM存储芯片构成4Kx8的存储器



4.3 存储器芯片的扩展（混合扩展）

例：4Kx4 SRAM存储芯片构成16Kx8的存储器

➤ 4Kx4 芯片：

- 12个地址管脚 A11~A0
- 4个数据管脚 D3~D0
- 1个片选输入管脚 CS#
- 1个读写控制管脚 WE#
- 芯片地址空间：000H~FFF H

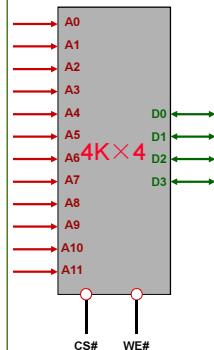
➤ CPU向存储器提供：

- 地址总线14根：AB13~AB0
- 数据总线8根：DB7~DB0
- 读写控制信号：MemW
- 存储器地址空间：0000H~3FFF H

➤ 需要芯片数：(16Kx8)/(4Kx4) = 8片

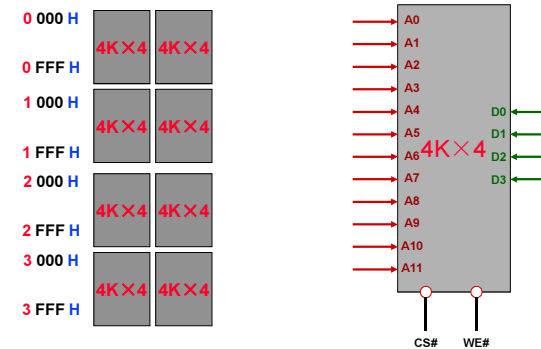
分4组（字扩展），每组2个芯片（位扩展）

- 一个2-4译码器产生4个片选信号
- 译码器输入：AB13~AB12
- 译码器输出：分别接4组芯片片选管脚



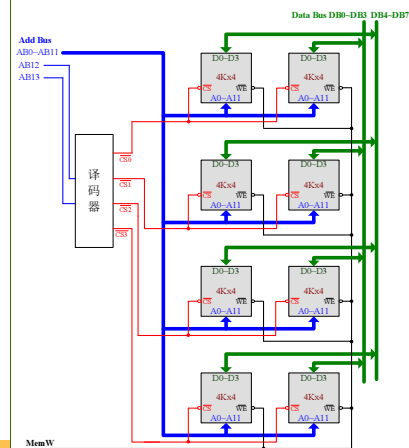
4.3 存储器芯片的扩展（混合扩展）

4Kx4 SRAM存储芯片构成16Kx8的存储器地址空间划分

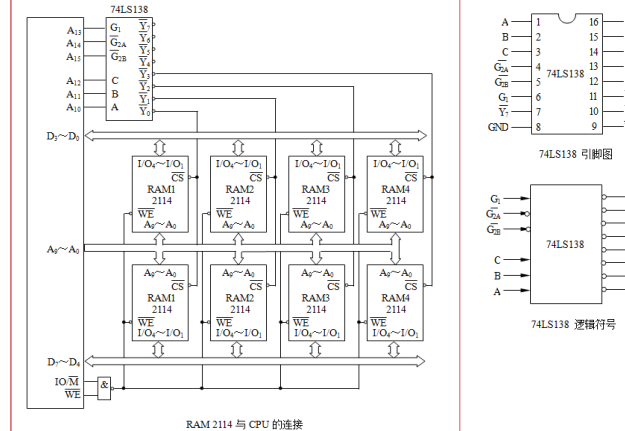


4.3 存储器芯片的扩展（混合扩展）

4Kx4 SRAM存
储芯片构成
16Kx8的存储
器连接图



存储器芯片的扩展示例（续）



存储器芯片的扩展方法小结

❖ 基本思路

1. 确定每个芯片的地址管脚数、数据管脚数
2. 确定整个存储空间所需的地址总线和数据总线的数量
3. 计算所需存储器芯片的数量，确定每个存储器芯片在整个存储空间中的地址空间范围、位空间范围
4. 所有**芯片的地址管脚**全部连接到**地址总线**对应的地址线上
5. **同一字空间的存储芯片CS信号**连在一起
6. **同一位空间的数据线**连在一起，并连接到对应的数据总线上
7. 根据每个存储器芯片的**地址空间范围**设计存储器芯片所需要的**片选信号逻辑**，CS逻辑电路的输入一定是**地址总线中没有连接到芯片的地址管脚上的那部分地址线**
8. 统一读写控制

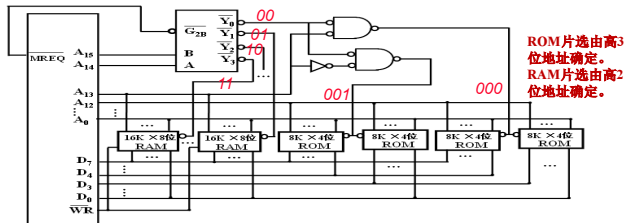
CPU与主存的连接（示例）

CPU地址线A15~A0，数据线D7~D0， \overline{WR} 为读/写信号， \overline{MREQ} 为访存请求信号。0000H~3FFFFH为系统程序区，4000H~FFFFH为用户程序区。用8K×4位ROM芯片和16K×8位RAM芯片构成该存储器，要求说明地址译码方案，并将ROM芯片、RAM芯片与CPU连接。

- ❖ 1. 确定每个芯片的地址管脚数、数据管脚数
 - ❖ 8K X 4 芯片: 13位地址，4位数据，芯片地址空间：0000 H ~ 1FFF H
 - ❖ 16K X 8 芯片: 14位地址，8位数据，芯片地址空间：0000 H ~ 3FFF H
- ❖ 2. 确定整个存储空间所需的地址总线和数据总线的数量
 - ❖ 16位地址，8位数据，地址空间：0000 H ~ FFFF H
 - 16位地址，8位数据，ROM，地址空间：0000H ~ 3FFFFH
 - 16位地址，8位数据，RAM，地址空间：4000H ~ FFFFFH
- ❖ 3. 计算所需存储器芯片的数量，确定每个存储器芯片在整个存储空间中的地址空间范围、位空间范围
 - ❖ 8K X 4 芯片：(16K X 8) / (8K X 4) = 4
 - ❖ 16K X 8 芯片：(48K X 8) / (16K X 8) = 3

CPU与主存的连接（示例-续）

4. 所有 芯片的地址管脚全部连接到地址总线对应的地址线上
5. 同一字空间的存储芯片 CS 信号连在一起
6. 同一位空间的数据线连在一起, 并连接到对应的数据总线上
7. 根据每个存储器芯片的地址空间范围设计存储器芯片所需要的片选信号逻辑
8. 统一读写控制



存储器的符号表示

❖ 读操作

► 输入

- 读单元地址: **Address**
- 读控制信号: **MemRead**

► **输出**

- 读出数据: **Readdata**

❖ 写操作

► 输入

- 写单元地址: **Address**
- 写入数据: **Writedata**
- 写控制信号: **MemWrite**

➤输出：无

